



UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH

Escola Tècnica Superior d'Enginyeria
de Telecomunicació de Barcelona

Inferència de la Topologia de Grafs

Tura Gimeno Sabater

Escola Tècnica Superior d'Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya

Supervisor Acadèmic: Alba Pagès-Zamora
Universitat Politècnica de Catalunya

En compliment parcial dels requisits per al títol
D'Enginyeria en Tecnologies i Serveis de Telecomunicació
Menció en *Sistemes Audiovisuals*

Agraïments

Vull agraïr a l'Alba Pagès Zamora tot l'ajut i acompanyament durant tot el procés del projecte. També vull donar les gràcies a la meva família i a l'Arnau Esteban, la meva parella, pel suport incondicional al llarg de tota la carrera.

Resum

Els grafs són una eina matemàtica que ens permet modelar les relacions entre dades obtingudes en estructures complexes, com per exemple el cervell, l'expressió dels genomes, xarxes de sensors, entre d'altres.

En aquest projecte s'investiguen tècniques de processament de senyal en grafs per inferir la topologia d'aquests a partir de dades observades. En particular es testegen diferents tècniques d'inferència de la topologia dels grafs usant dades reals. Un dels principals objectius va consistir en trobar una base de dades idònia per a realitzar l'experiment. Aquesta cerca ens va portar fins a una base de dades d'EEG's (Electroencefalogrames) d'un cervell humà. Com a tasca addicional també s'ha estudiat la utilitat de la topologia dels grafs resultants per a classificar les dades.

Abstract

Graphs are a mathematical tools that allow us to model relationships between data obtained in complex structures, such as the brain, expression of genomes, sensor networks, and more.

This project investigates signal processing techniques in graphs to infer the topology of them from observed data. In particular, different inference techniques of graph topology are tested using real data. One of the main goals was to find a suitable database for the experiment. This search led us to a database of EEG's (Electroencephalograms) in a human brain. In addition, it has been studied the usefulness of the resulting topology of the graphs to classify the data.

Resumen

Los grafos son una herramienta matemática que nos permite modelar las relaciones entre datos obtenidos en estructuras complejas, como por ejemplo el cerebro, la expresión de los genomas, redes de sensores, entre otros.

En este proyecto se investigan técnicas de procesamiento de señal en grafos para inferir la topología de éstos a partir de datos observados. En particular se testean diferentes técnicas de inferencia de la topología de los grafos usando datos reales. Uno de los principales objetivos consistió en encontrar una base de datos idónea para realizar el experimento. Esta búsqueda nos llevó hasta una base de datos de EEG's (Electroencefalogramas) de un cerebro humano. Como tarea adicional también se ha estudiado la utilidad de la topología de los grafos resultantes para clasificar los datos.

######

Índex

1	Introducció	1
1.1	Objectius	1
1.2	Requeriments i especificacions	1
1.3	Mètodes i procediments	2
1.4	Pla de treball	2
1.4.1	Paquets de Treball	2
1.4.2	Diagrama de Gantt	3
1.5	Incidències i modificacions	3
2	Inferència de la topologia de grafs a partir de dades	4
2.1	Base teòrica dels grafs i les seves aplicacions	4
2.1.1	Definició de graf i matrius associades	4
2.1.2	Matriu Laplaciana	7
2.1.3	Matriu Laplaciana normalitzada	8
2.2	Mètodes per inferir la topologia del graf	9
2.2.1	Mètode de correlació de Pearson	9
2.2.2	Mètode de selecció de covariància	10
2.3	Aplicació de la topologia de grafs a la classificació de senyals	13

2.3.1	Base de dades	13
2.3.2	Classificació a partir de la matriu Laplaciana del graf inferit	14
3	Resultats	16
3.1	Resultats per un sol arxiu	16
3.2	Utilització dels autovalors de les Laplacianes per la classificació	17
3.3	Resultats generals	19
4	Pressupost	20
4.1	Llicència	20
4.2	Salaris	20
5	Conclusions i possibles millores	21
6	Apèndixs	24

Índex de figures

1.1	Diagrama de Gantt del Projecte	3
2.1	Representació graf	5
2.2	Tipus de grafs	5
2.3	Graf amb pesos	6
2.4	Graf pel càlcul de la Laplaciana	7
2.5	Graf matriu covariància empírica	10
2.6	Graf xarxa correlació Pearson	10
2.7	Distribució punts del cervell	13

Índex de taules

2.1	Exemple arxiu base de dades	13
2.2	Classificació imatges per grups	14
3.1	Taula matrius	17
3.2	Taula autovalors arxiu	17
3.3	Taula autovalors grups	18
3.4	Matrius de confusió	19

Capítol 1

Introducció

En aquest primer capítol s'especifiquen els objectius del projecte i es detallen les fases en les quals s'ha dividit l'estudi. Durant la investigació, el plantejament inicial del treball ha anat evolucionant segons els resultats obtinguts. És per aquest motiu, que el primer apartat també inclou incidències i modificacions que han anat sorgint durant l'elaboració del projecte.

1.1 Objectius

En un inici, es van marcar uns objectius concrets que s'han anat modificant i ampliant a mesura que ha anat avançant la investigació. Els objectius són els següents:

- ◇ Estudiar la teoria d'àlgebra de grafs i les seves aplicacions.
- ◇ Aprendre tècniques de processament de senyal en grafs per a inferir la topologia d'un graf a partir de dades observades.
- ◇ Programar en Matlab els mètodes per a inferir grafs a partir de dades observades.
- ◇ Cercar una base de dades apropiada per a avaluar els mètodes programats amb dades reals, i investigar una possible aplicació.

1.2 Requeriments i especificacions

Tenint en compte els objectius esmentats anteriorment, el principal requeriment del projecte és classificar correctament els EEG's extrets del cervell d'un subjecte mentre mira una sèrie d'imatges utilitzant el graf inferit. Per inferir la topologia d'aquests grafs s'han d'utilitzar els mètodes proposats en el projecte, i l'algorisme de classificació ha de poder classificar un alt percentatge de dades.

D'altra banda, les especificacions del treball són utilitzar una base de dades real, utilitzar Matlab com a eina matemàtica i utilitzar els conceptes matemàtics apresos durant el projecte per implementar l'algorisme final.

1.3 Mètodes i procediments

Les dades utilitzades en aquest estudi han estat extretes de la base de dades pública “MindBigData IMAGENET of The Brain” [1], explicada en l’apartat 2.3.1. Aquesta base de dades conté una sèrie d’EEG’s emmagatzemats en arxius **.csv**, on hi ha el recull d’informació extreta des de 5 punts del cervell. Cada arxiu correspon a un registre de 3 segons del subjecte mirant una imatge concreta. S’ha de tenir en compte que aquesta base de dades no s’ha elaborat expressament per utilitzar-la en un experiment com el que s’ha dut a terme.

Aquest projecte s’ha realitzat amb Matlab com a eina principal. S’ha utilitzat una gran quantitat d’informació sobre grafs, aplicacions de grafs i mètodes per inferir la topologia de grafs. El codi desenvolupat es troba a l’Apèndix (6).

1.4 Pla de treball

El projecte s’ha planificat en diverses fases, especificades a l’apartat 1.4.1. Com que durant l’investigació el pla de treball s’ha anat modificant, el que es presenta a continuació es l’última versió.

1.4.1 Paquets de Treball

Els paquets de treball d’aquest projecte són 4:

- ◇ Documentació.
- ◇ Recerca de la informació i investigació.
- ◇ Desenvolupament del projecte.
- ◇ Presentació del projecte.

Aquests paquets de treball es detallen al diagrama de Gantt de la secció 1.4.2 on també es mostren les tasques concretes de cada paquet.

1.4.2 Diagrama de Gantt

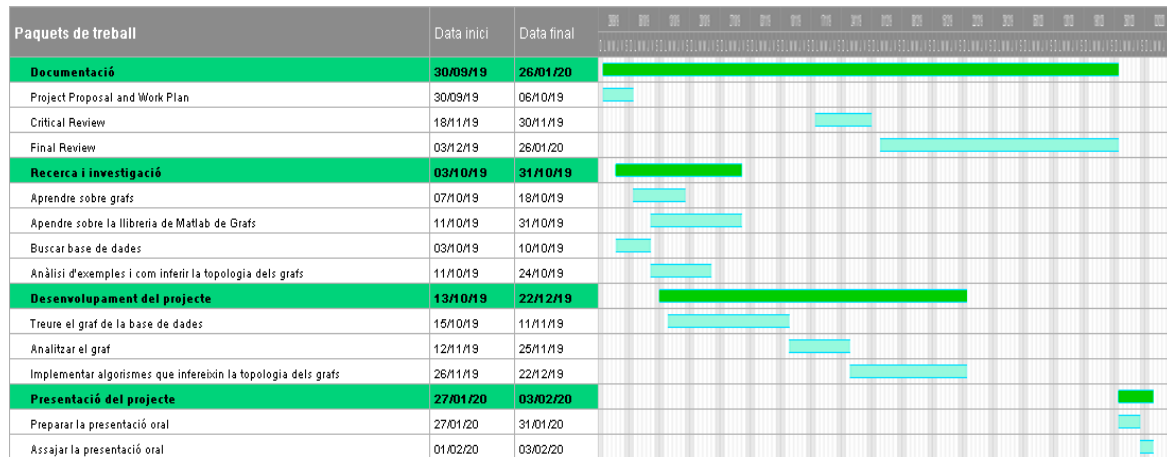


Figura 1.1: Diagrama de Gantt del projecte

1.5 Incidències i modificacions

Com hem esmentat en l'apartat 1.1, un dels objectius es basa en investigar tècniques per inferir la topologia dels grafs a partir de dades utilitzant diferents mètodes. La principal qüestió és si això es podria fer a partir de dades reals, i concretament, si es podria fer servir per classificar aquestes dades.

Un dels reptes importants ha estat trobar una base de dades que permeti obtenir grafs a partir de les dades. Després d'una fase de cerca, la base de dades escollida va ser "MindBig-Data IMAGENET of The Brain" [1]. L'únic inconvenient és que són grafs de només 5 nodes, cosa que pot influir de manera negativa a l'hora de classificar els senyals.

"MindBigData IMAGENET of The Brain" recull un total de 10.032 imatges i 10.032 arxius. Gestionar aquest volum d'informació ha estat un altre dels reptes, ja que tantes dades han dificultat la creació de l'algorisme d'inferència del graf. Davant aquesta situació, es va decidir utilitzar només una part de la base de dades.

Una altra de les dificultats ha estat trobar el mètode idoni per la classificació dels EEG's. Això ha portat a diferents modificacions respecte a com inferir la topologia dels grafs obtinguts de la base de dades.

Capítol 2

Inferència de la topologia de grafs a partir de dades

La part més important d'aquest projecte és l'estudi dels grafs i com inferir la topologia d'aquests. La inferència d'aquests grafs s'utilitzarà per classificar EEG's (Electroencefalogrames). Aquesta classificació es farà amb les dades obtingudes dels EEG's del cervell d'una persona.

Per inferir la topologia dels grafs hi ha diversos mètodes. Concretament es farà referència a dos d'ells.

També es parlarà sobre la base de dades utilitzada i quin ha estat el procés per a l'experimentació del projecte.

2.1 Base teòrica dels grafs i les seves aplicacions

En aquesta secció es parla del concepte de graf, i es presenta la teoria d'àlgebra de grafs utilitzada en el projecte. La informació sobre tots aquests conceptes ha estat obtinguda de la teoria del seminari de l'ETSETB *Graph Signal Processing* [2]. La part teòrica va acompanyada d'exemples.

2.1.1 Definició de graf i matrius associades

Un **graf** està definit per $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. \mathcal{V} és el conjunt **nodes** indexat amb $i = \{1, \dots, N\}$. $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ és un conjunt ordenat de parells de nodes de \mathcal{V} anomenats **arestes** que representa una connexió entre dos nodes.

Per poder veure de manera gràfica que és un graf, la figura 2.1 ens mostra un graf de 5 nodes $\{1, 2, 3, 4, 5\}$ i 6 arestes que connecten els nodes entre ells.

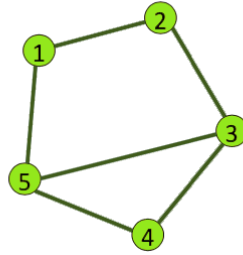


Figura 2.1: Representació graf de 5 nodes i 6 arestes

Una aresta (i, j) entre els nodes i i j es refereix a la informació que va des del node j al node i . Si s'assigna una direcció a les arestes, les relacions passen a ser asimètriques. Aquest tipus de graf s'anomena graf dirigit o digraf. Si no s'assignen direccions a les arestes, llavors $(i, j) \in \mathcal{E} \Leftrightarrow (j, i) \in \mathcal{E}$ per totes les $i, j \in \mathcal{V}$ i el graf s'anomena graf no dirigit. En la següent figura es mostra un exemple de cada tipus de graf.

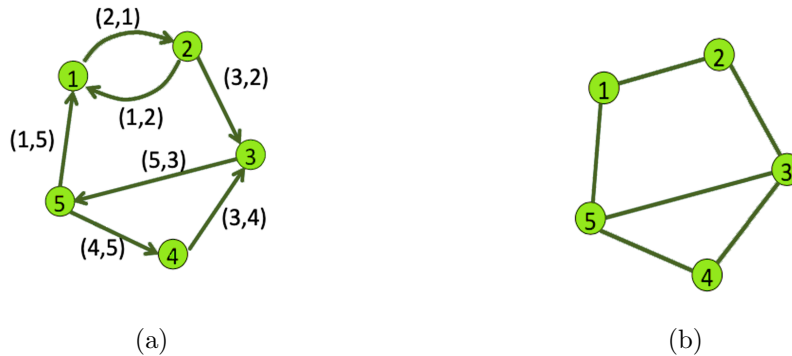


Figura 2.2: Tipus de grafs: (a) Graf dirigit i (b) Graf no dirigit

El conjunt de veïns del **node** i es defineix com a:

$$\mathcal{N}_i := \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$$

El grau d'un node és el nombre d'arestes entrant en aquell node, donat per $\deg(i) = |\mathcal{N}_i|$. En els grafs dirigits es pot diferenciar entre grau d'entrada i de sortida dels nodes. També és important el concepte del pes associat a cada aresta, com podem observar en la figura 2.3. Aquests pesos s'associen segons un mapa $W : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}_+$ tal que:

$$W(i, j) = \begin{cases} \geq 0 & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{altres} \end{cases}$$

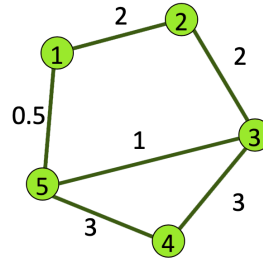


Figura 2.3: Graf amb pesos

La matriu d'adjacència de \mathcal{G} es defineix com $\mathbf{A} \in \mathbb{R}^{N \times N}$, una matriu amb entrades a_{ij} donades per:

$$a_{ij} = \begin{cases} W(i, j) & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{altres} \end{cases}$$

La matriu d'adjacència del graf no dirigit vist anteriorment en la figura 2.3 seria la següent:

$$\mathbf{A} = \begin{bmatrix} 0 & 2 & 0 & 0 & 0.5 \\ 2 & 0 & 2 & 0 & 0 \\ 0 & 2 & 0 & 3 & 1 \\ 0 & 0 & 3 & 0 & 3 \\ 0.5 & 0 & 1 & 3 & 0 \end{bmatrix}$$

La **matriu de graus** \mathbf{D} d'un graf $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$ amb N nodes està definida com:

$$\mathbf{D} = \text{diag}(\mathbf{A} \cdot \mathbf{1})$$

on $\mathbf{1}$ és un vector d'uns, i $\text{diag}(\mathbf{v})$ és una matriu diagonal on les components de la diagonal principal són les entrades de \mathbf{v} . La següent matriu \mathbf{D} és la **matriu de graus** del graf de la figura 2.3:

$$\mathbf{D} = \begin{bmatrix} 2.5 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 4.5 \end{bmatrix}$$

2.1.2 Matriu Laplaciana

Donat un graf no dirigit $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$ amb N nodes, la **Laplaciana** $L \in \mathbb{R}^{N \times N}$ es defineix com:

$$L = D - A$$

$D = \text{diag}(A \cdot \mathbf{1})$, i A es la matriu d'adjacència.

Les entrades de la Laplaciana són iguals a:

$$l_{ij} = \begin{cases} d_{ii} & \text{if } i = j \\ -a_{ij} & \text{if } i \neq j \end{cases}$$

Utilitzant el graf de la figura 2.4 i els seus pesos, calcularem la seva laplaciana.

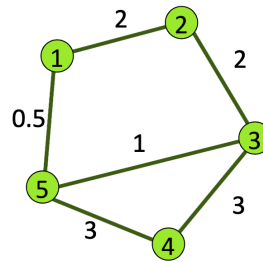


Figura 2.4: Graf pel càlcul de la laplaciana

$$A = \begin{bmatrix} 0 & 2 & 0 & 0 & 0.5 \\ 2 & 0 & 2 & 0 & 0 \\ 0 & 2 & 0 & 3 & 1 \\ 0 & 0 & 3 & 0 & 3 \\ 0.5 & 0 & 1 & 3 & 0 \end{bmatrix}$$

$$D = \text{diag}(A \cdot \mathbf{1}) = \begin{bmatrix} 2.5 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 6 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 4.5 \end{bmatrix}$$

$$\mathbf{L} = \mathbf{D} - \mathbf{A} = \begin{bmatrix} 2.5 & -2 & 0 & 0 & -0.5 \\ -2 & 4 & -2 & 0 & 0 \\ 0 & -2 & 6 & -3 & -1 \\ 0 & 0 & -3 & 6 & -3 \\ -0.5 & 0 & -1 & -3 & 4.5 \end{bmatrix}$$

La matriu Laplaciana d'un graf \mathbf{L} és simètrica i semi-definida positiva

$$\mathbf{x}^T \mathbf{L} \mathbf{x} = \frac{1}{2} \sum_{\forall i,j} a_{ij} \cdot (x_i - x_j)^2 = \sum_{(i,j) \in \mathcal{E}} a_{ij} \cdot (x_i - x_j)^2; \quad \forall \mathbf{x} \in \mathbb{R}^{N \times 1}$$

Per tant, els seus autovalors $\{\lambda_i\}_{i=1}^N$ són reals i no negatius.

S'ha de tenir en compte que \mathbf{L} té sempre un autovalor igual a 0 amb autovector $\mathbf{1}$.

El teorema de Geršgorin estableix que els autovalors de qualsevol matriu \mathbf{L} es troben en la unió dels N discs de Geršgorin.

$$\bigcup_{i=1}^N \{z \in \mathbb{C} : |z - l_{ii}| \leq R_i\}$$

on $R_i = \sum_{\substack{j=1 \\ j \neq i}}^N |l_{ij}|$. Per tant, $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N \leq 2d_{\max}$.

Els autovalors de la matriu Laplaciana seran utilitzats posteriorment per classificar els arxius, explicat en el punt 2.3.2.

2.1.3 Matriu Laplaciana normalitzada

Donat un graf no dirigit $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, W\}$ amb N nodes, la **Laplaciana Normalitzada** $\mathbf{L}^{sn} \in \mathbb{R}^{N \times N}$ es defineix com:

$$\mathbf{L}^{sn} = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}} \mathbf{A} \mathbf{D}^{-\frac{1}{2}}$$

On \mathbf{A} és la matriu d'ajacència, $\mathbf{D} = \text{diag}(\mathbf{A} \cdot \mathbf{1})$, \mathbf{I} és la matriu identitat i \mathbf{L} és la matriu de la Laplaciana.

Cada entrada d'aquesta matriu està definida com:

$$l_{ij}^{sn} = \begin{cases} 1 & \text{if } i = j \\ \frac{-a_{ij}}{\sqrt{d_{ii}d_{jj}}} & \text{if } (i, j) \in \mathcal{E} \\ 0 & \text{altres} \end{cases},$$

Els seus autovalors estan limitats $0 = \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N \leq 2$, i

$$\mathbf{x}^T \mathbf{L}^{sn} \mathbf{x} = \sum_{(i,j) \in \mathcal{E}} a_{ij} \cdot \left(\frac{x_i}{\sqrt{d_{ii}}} - \frac{x_j}{\sqrt{d_{jj}}} \right)^2$$

Els autovalors de la matriu Laplaciana normalitzada seran utilitzats posteriorment per classificar els arxius, explicat en el punt 2.3.2.

2.2 Mètodes per inferir la topologia del graf

Partint de la base que tenim un conjunt de vectors d'observació $\mathcal{X} := \{\mathbf{x}_p\}_{p=1}^P$ independents, s'analitzen dos mètodes per poder inferir la topologia del graf: mètode de correlació de Pearson i mètode de selecció de covariància. Inferir la topologia d'un graf fa referència al problema fonamental de com utilitzar observacions de dades per aprendre l'estructura gràfica adjacent.

2.2.1 Mètode de correlació de Pearson

Una mesura lineal àmpliament adoptada entre les variables nodals x_i i x_j és el coeficient de correlació Pearson definit com:

$$\rho_{ij} = \frac{\text{cov}(x_i, x_j)}{\sqrt{\text{var}(x_i)\text{var}(x_j)}} \quad (2.1)$$

que es pot obtenir de les entrades $\sigma_{ij} := \text{cov}(x_i, x_j)$ a la matriu de covariància $\mathbf{\Sigma} := \mathbb{E}[(\mathbf{x} - \boldsymbol{\mu})(\mathbf{x} - \boldsymbol{\mu})^T]$ and $\boldsymbol{\mu} := \mathbb{E}[\mathbf{x}]$.

Tenint en compte el conjunt de vectors d'observació independents \mathcal{X} , es construeix la matriu de covariància empírica (2.2):

$$\hat{\mathbf{\Sigma}} := \frac{1}{P} \sum_{p=1}^P \mathbf{x}_p \mathbf{x}_p^T \quad (2.2)$$

Aquesta matriu de covariància empírica serà la matriu d'adjacència del graf de la figura 2.5. Com es pot observar els pesos del graf seran covariàncies.

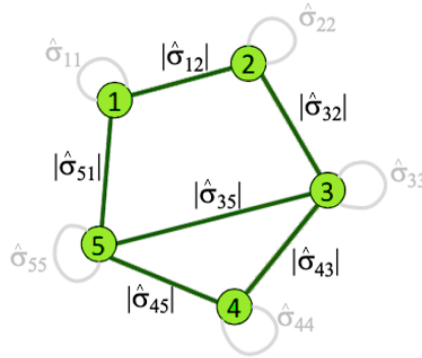


Figura 2.5: Graf obtingut a partir de la matriu de covariància empírica

D'aquesta manera, utilitzant les covariàncies obtingudes i l'equació (2.1) del coeficient de pearson, s'obté la matriu de correlació de Pearson. Aquesta matriu es la matriu d'adjacència del graf de correlació de Pearson (figura 2.6). Els llindars solen aplicar-se de manera que els talls s'assignen als valors més importants corresponents de $|\rho_{ij}|$.

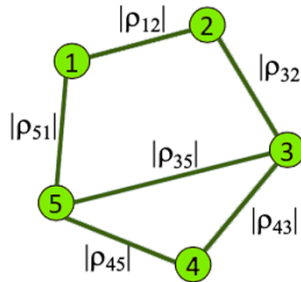


Figura 2.6: Graf correlació de Pearson

2.2.2 Mètode de selecció de covariància

Considerem que \mathbf{x} és un vector aleatori Gaussià $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

Aleshores, $\rho_{ij}|\mathcal{V}_{ij} = 0$ equival a x_i i x_j sent condicionalment independents, i. e.

$$x_i \perp x_j | \mathbf{x}_{-ij}$$

El *Gaussian Markov Random Field* (GMRF) és un vector aleatori Gaussià $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ amb propietats de Markov

$$x_i \perp x_j | \mathbf{x}_{-ij}$$

x_i i x_j son condicionalment independents.

Els GMRFs poden definir-se en termes de *matriu de presició*, com $\boldsymbol{\Theta} := \boldsymbol{\Sigma}^{-1}$.

$$f(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Theta}) \propto |\boldsymbol{\Theta}|^{1/2} \exp \left(-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Theta} (\mathbf{x} - \boldsymbol{\mu}) \right)$$

On es pot demostrar que:

$$x_i \perp x_j | \mathbf{x}_{-ij} \iff \theta_{ij} = 0$$

Sota aquest model, l'estimació de la màxima versemblança (ML) de la matriu de precisió es troba de la següent manera:

S'ha eliminat la mitjana de les dades de manera que podem suposar que $\boldsymbol{\mu} = 0$ i que les dades observades són independents.

$$f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p; \boldsymbol{\Theta}) = \prod_{p=1}^P f(\mathbf{x}_p; \boldsymbol{\Theta}) \quad (2.3)$$

$$= \prod_{p=1}^P \frac{|\boldsymbol{\Theta}|^{1/2}}{(2\pi)^N} \exp \left(-\frac{1}{2} \mathbf{x}_p^T \boldsymbol{\Theta} \mathbf{x}_p \right) \quad (2.4)$$

$$= \left(\frac{1}{\sqrt{(2\pi)^N}} \right)^P |\boldsymbol{\Theta}|^{P/2} \prod_{p=1}^P \exp \left(-\frac{1}{2} \mathbf{x}_p^T \boldsymbol{\Theta} \mathbf{x}_p \right) \quad (2.5)$$

Per calcular l'estimador ML, utilitzem la fórmula de l'argument màxim de la funció 2.3:

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_{ML} &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p; \boldsymbol{\Theta}) \\ &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} \{ \ln f(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p; \boldsymbol{\Theta}) \} \\ &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} \left\{ -\frac{NP}{2} \ln 2\pi + \frac{P}{2} \ln |\boldsymbol{\Theta}| - \sum_{p=1}^P \frac{1}{2} \mathbf{x}_p^T \boldsymbol{\Theta} \mathbf{x}_p \right\} \\ &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} \left\{ \frac{P}{2} \ln |\boldsymbol{\Theta}| - \frac{1}{2} \sum_{p=1}^P \mathbf{x}_p^T \boldsymbol{\Theta} \mathbf{x}_p \right\} \\ &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} \left\{ \frac{P}{2} \ln |\boldsymbol{\Theta}| - \frac{1}{2} \operatorname{Tr}(\mathbf{X}^T \boldsymbol{\Theta} \mathbf{X}) \right\} \\ &= \underset{\boldsymbol{\Theta} \succeq \mathbf{0}}{\operatorname{argmax}} \left\{ \frac{P}{2} \ln |\boldsymbol{\Theta}| - \frac{1}{2} \operatorname{Tr}(\boldsymbol{\Theta} \mathbf{X} \mathbf{X}^T) \right\} \end{aligned}$$

On $\operatorname{Tr}()$ denota la traça de una matriu.

$$\mathbf{X}\mathbf{X}^T = \sum_{p=1}^P \mathbf{x}_p \mathbf{x}_p^T = P\hat{\Sigma} \quad (2.6)$$

Utilitzant aquesta igualtat (2.6) s'obté que l'estimador ML de la matriu de precisió és el següent:

$$\begin{aligned} \hat{\Theta}_{ML} &= \underset{\Theta \succeq 0}{\operatorname{argmax}} \left\{ \frac{P}{2} \ln |\Theta| - \frac{P}{2} \operatorname{Tr}(\Theta \hat{\Sigma}) \right\} \\ &= \underset{\Theta \succeq 0}{\operatorname{argmax}} \left\{ \ln |\Theta| - \operatorname{Tr}(\Theta \hat{\Sigma}) \right\} \end{aligned}$$

Per calcular l'argument màxim fem la derivada en funció del que es vol estimar, i s'iguali a 0. D'aquesta manera es calcula l'estimador ML final.

$$\frac{d}{d\Theta} \{ \ln |\Theta| - \operatorname{Tr}(\Theta \hat{\Sigma}) \} = 0 \quad (2.7)$$

Utilitzem les següents igualtats [3] per calcular les derivades:

$$\frac{d}{d\Theta} \operatorname{Tr}(\Theta \hat{\Sigma}) = \hat{\Sigma}^T$$

$$\frac{d}{d\Theta} \ln |\Theta| = \Theta^{-T}$$

Amb aquestes igualtats podem aïllar l'estimador de la fórmula anterior (2.7):

$$\Theta^{-T} - \hat{\Sigma}^T = 0$$

$$\Theta^{-T} = \hat{\Sigma}^T$$

I aquesta és la fórmula final de l'estimador ML:

$$\hat{\Theta}_{ML} = \hat{\Sigma}^{-1}$$

2.3 Aplicació de la topologia de grafs a la classificació de senyals

Per tal d'avaluar els mètodes descrits anteriorment, s'agafa una base de dades, i utilitzant Matlab, es classifiquen els EEG's del cervell segons la imatge que està veient el subjecte. A continuació s'explica quina és la base de dades utilitzada i com s'ha fet l'experimentació.

2.3.1 Base de dades

La base de dades utilitzada per fer aquest projecte es "MindBigData IMAGENET of The Brain"[1]. Aquesta base de dades conté 10.032 senyals cerebrals de 3 segons cadascuna, capturades amb l'estímul de veure una imatge aleatòria del conjunt de dades Imagenet ILSVRC2013".

Disposem de 10.032 arxius *.csv que contenen els senyals cerebrals des de diferents punts del cervell, quan el subjecte veu una imatge. Cada arxiu correspon a l'enregistrament d'una sola imatge. Els punts del cervell des d'on en capturen els diferents senyals són: "AF3", "AF4", "T7", "T8", "Pz".

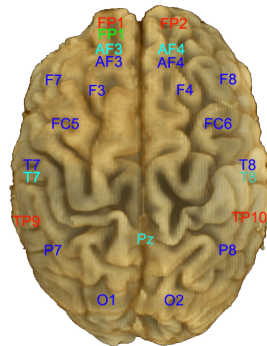


Figura 2.7: Distribució punts del cervell [1]

Les dades estan en forma de taules de valors. Hi ha tantes columnes com senyals capturats durant 3 segons, i tantes files com punts en el cervell. A la taula 2.1 s'observa part d'un dels arxius *.csv que hi ha a la base de dades. Concretament de l'arxiu MindBigData-Imagenet-Insight-n01496331-15294-1-1968.csv.

AF3	4274,359	4276,41	4278,974	4278,974	4276,923	4275,897	4275,385	4279,487
AF4	4263,59	4248,718	4227,692	4243,077	4269,231	4252,308	4240	4253,333
T7	4261,538	4262,051	4255,385	4256,41	4263,077	4264,103	4264,615	4268,205
T8	4165,641	4187,692	4203,59	4192,821	4186,154	4185,641	4186,154	4198,974
Pz	4161,538	4163,59	4160,513	4164,615	4175,897	4178,462	4173,846	4178,462

Taula 2.1: Part de l'arxiu MindBigData-Imagenet-Insight-n01496331-15294-1-1968.csv

La base de dades està organitzada en diferents grups, i per cada grup hi ha varies imatges. El grup depèn de la imatge que veu el subjecte. A la taula 2.2 tenim un exemple de com s'organitza:

IMATGES	NUMERO IMATGES	NOM ARXIU
person, individual, someone, somebody, mortal, soul	17	n00007846
check-in	11	n00141669
polo	16	n00477639
goldfish, Carassius auratus	12	n01443537
ray	25	n01495701
electric ray, crampfish, numbfish, torpedo	14	n01496331
stingray	21	n01498041
bird	17	n01503061
cock	18	n01514668
hen	11	n01514859
ostrich, Struthio camelus	18	n01518878
brambling, Fringilla montifringilla	21	n01530575
goldfinch, Carduelis carduelis	16	n01531178
house finch, linnet, Carpodacus mexicanus	16	n01532829
juncos, snowbird	18	n01534433
indigo bunting, indigo finch, indigo bird, Passerina cyanea	24	n01537544

Taula 2.2: Classificació de les imatges per grups

D'aquests grups, se n'han tret 4 grups generals:

- ◇ **Animals Marins** - 109 imatges
- ◇ **Serps** - 282 imatges
- ◇ **Gossos** - 252 imatges
- ◇ **Piano** - 56 imatges

Aquests 698 arxius seran els utilitzats per creat l'algorisme de classificació. La classificació es farà només en 4 grups, ja que era molt complicat classificar 10.032 imatges.

2.3.2 Classificació a partir de la matriu Laplaciana del graf inferit

L'experimentació s'ha fet amb la base de dades explicada anteriorment. Com ja s'ha comentat, s'han agafat 4 grups per fer aquest experiment. De cada un dels grups, s'ha calculat la mitjana de cadascun dels 5 autovalors de les Laplacianes ($\lambda_{(grup)}$). També s'han calculat els autovalors de les Laplacianes (λ) de l'EEG que es vol classificar.

A l'hora de classificar un EEG segons el grup s'utilitza la fórmula (2.8) que és l'error quadràtic dels autovalors respecte de la mitjana dels autovalors del grup. On λ_i és l'autovalor en la posició i del EEG que volem classificar, i $\lambda_{i(grup)}$ és l'autovalor en la posició i del vector d'autovalors del grup. D'aquí treure'm 4 valors, un per cada grup.

$$\sum (\lambda_i - \lambda_{i(grup)})^2 \quad (2.8)$$

D'aquests 4 valors trobats, el que sigui més petit, serà el que marcarà quin és el grup al qual pertany l'EEG en qüestió. Aquest és un dels mètodes pensats per a poder classificar.

Per fer aquest experiment s'ha utilitzat el Matlab com a eina principal. El codi calcula totes les dades necessàries per calcular els grafs, les Laplacianes i els autovalors de totes les imatges. Els arxius que agafa el codi són els de la base de dades, més alguns creats expressament amb la classificació de les llistes per grups. Els codis de l'experimentació el trobareu a l'Apèndix (6).

Capítol 3

Resultats

En aquest apartat es mostren els resultats obtinguts de la fase d'experimentació del projecte. Per mostrar-los d'una forma clara, es presentarà el resultat de passar pel codi Matlab (6) un dels arxius de la base de dades, en forma de matrius/vectors i resultats numèrics.

Es mostraran els resultats només d'un arxiu, i després es mostrarà l'estudi dels resultats en general. És important veure en el cas d'un exemple quins són els paràmetres que es van obtenint.

De cadascun dels arxius de la base de dades, s'utilitzen les dades de cada columna com a vectors d'observació. Els 5 punts del cervell seran els 5 nodes del graf.

Cada vector d'observació \mathbf{x}_p , és la informació en un instant de temps, de l'EEG en 5 punts diferents d'un cervell humà, i el conjunt \mathcal{X} , fa referència al conjunt de vectors \mathbf{x}_p en certs instants de temps durant 3 segons. Aquest vector d'observació és el que s'utilitzarà per fer tots els càlculs explicats anteriorment (2.2).

3.1 Resultats per un sol arxiu

L'arxiu que s'utilitza per a la mostra de resultats és el *MindBigData – Imagenet – Insight – n02084071 – 12317 – 1 – 954.csv*. Aquest arxiu forma part del grup de "Gossos" anomenat anteriorment al punt 2.3.1.

Un cop es té clar quin arxiu s'utilitza i quina és la informació que utilitzem d'aquest, passem l'arxiu pel codi Matlab que s'ha creat per a l'experimentació. Els resultats obtinguts són els següents:

Taula 3.1: Taula amb les matrius resultants

Matrius d'adjacència amb els dos mètodes estudiats									
Matriu d'adjacència mètode de correlació de Pearson					Matriu d'adjacència mètode de selecció de covariància				
0	0.8663	0.4122	0.6207	0.3433	0	0.0058	0.00018	0.00017	0.0013
0.8663	0	0.5015	0.7556	0.5059	0.0058	0	0.0008	0.0035	0.0015
0.4122	0.5015	0	0.3355	0.1261	0.00018	0.0008	0	0	0.0003
0.6207	0.7556	0.3355	0	0.6953	0.00017	0.0035	0	0	0.0043
0.3433	0.5059	0.1261	0.6953	0	0.0013	0.0015	0.0003	0.0043	0
Matrius Laplacianes amb el mètode de selecció de covariància									
Matriu Laplaciana					Matriu Laplaciana Normalitzada				
0.0074	-0.0058	-0.0002	-0.0002	-0.0013	1	-0.6204	-0.0535	-0.0222	-0.1737
-0.0058	0.0117	-0.0008	-0.0035	-0.0015	-0.6204	1	-0.2128	-0.3609	-0.1634
-0.0002	-0.0008	0.0015	0	-0.0003	-0.0535	-0.2128	1	-0.0171	-0.1122
-0.0002	-0.0035	0	0.0080	-0.0043	-0.0222	-0.3609	-0.0171	1	-0.5531
-0.0013	-0.0015	-0.0003	-0.0043	0.0075	-0.1737	-0.1634	-0.1122	-0.5531	1

Com s'observa en les taules, només es calcula la matriu Laplaciana i Laplaciana normalitzada dels valors obtinguts amb el mètode de selecció de covariància. Això és perquè en el cas del mètode de correlació de Pearson els resultats de les Laplacianes no ajudaven a discriminar els senyals. Per tant la matriu de covariància empírica només es fa servir pel mètode de selecció de covariància.

3.2 Utilització dels autovalors de les Laplacianes per la classificació

Volem fer servir aquestes topologies per classificar, i com es pot observar en l'apartat anterior, els grafs no discriminen entre senyals. És per això que es decideix utilitzar els autovalors de les Laplacianes.

Taula 3.2: Recull dels autovalors obtinguts com a resultats del experiment (Arxiu)

Autovalors de l'arxiu que volem classificar					
Laplaciana	0	0.0018	0.0060	0.0114	0.0167
Laplaciana Normalitzada	0	0.7502	1.0300	1.4202	1.7993

També es necessiten les mitjanes de cada autovalor de la Laplaciana i la Laplaciana normalitzada de cada grup. Això es necessita per calcular l'error quadràtic.

Taula 3.3: Recull dels autovalors obtinguts com a resultats del experiment (Grups)

Mitjana de cada autovalor de la Laplaciana					
Animals Marins	0	0.0029	0.0047	0.0073	0.0111
Serps	0	0.0028	0.0049	0.0074	0.0117
Gossos	0	0.0026	0.0046	0.0071	0.0112
Piano	0	0.0030	0.00491	0.0076	0.0121
Mitjana de cada autovalor de la Laplaciana normalitzada					
Animals Marins	0	0.8412	1.1235	1.3812	1.6539
Serps	0	0.8361	1.1134	1.3905	1.6598
Gossos	0	0.8336	1.1209	1.3854	1.6599
Piano	0	0.8682	1.1066	1.3928	1.6323

En aquest punt es calcula l'error quadràtic (EQ) (2.8) de cadascun dels grups i es col·loca en un vector que té la següent estructura:

$$EQGrups = \begin{bmatrix} EQAnimalsMarins & EQSerps & EQGossos & EQPiano \end{bmatrix} \quad (3.1)$$

Els resultats obtinguts de l'error quadràtic de la Laplaciana per cada grup són:

$$EQLap = \begin{bmatrix} 0 & 4.3226e-05 & 5.2087e-05 & 3.8297e-05 \end{bmatrix} \quad (3.2)$$

Error quadràtic de la Laplaciana normalitzada per cada grup:

$$EQLapNorm = \begin{bmatrix} 0.0396 & 0.0346 & 0.0358 & 0.0484 \end{bmatrix} \quad (3.3)$$

Observem que dels valors dels vectors EQLap (3.2) i EQLapNorm (3.3) hi ha un valor que és el més petit. Aquest valor més petit serà l'error quadràtic del grup que l'algorisme classifica com a correcte (3.1). És a dir:

- ◇ Segons la Laplaciana - grup 4 (Piano)
- ◇ Segons la Laplaciana normalitzada - grup 2 (Serps)

Com s'observa, en aquest cas el classificador no funciona correctament.

3.3 Resultats generals

Per poder tenir una visió global dels resultats de la resta d'arxius, utilitzem les matrius de confusió [4]. La matriu de confusió funciona de la següent manera: cada columna de la matriu representa el nombre de prediccions de cada classe, mentre que cada fila representa les instàncies en la classe real. Per tant el nombre d'arxius que classifica correctament són els que es troben a la diagonal. Els resultats de la matriu de confusió de l'experimentació són:

Taula 3.4: Matrius de confusió

Matriu de confusió Laplaciana				Matriu de confusió Laplaciana normalitzada			
9	1	54	45	19	25	18	47
22	7	148	105	34	77	52	119
18	5	136	93	47	54	50	100
7	1	24	24	5	12	5	34

Aquí s'observa que classifica molts arxius malament. Si es parla de percentatge d'encerts, s'obté que en el cas de la Laplaciana hi ha un **25.2149** % d'encerts, i en el cas de la Laplaciana normalitzada un **25.7880** % d'encerts.

Capítol 4

Pressupost

En aquest capítol es calcula el pressupost total d'aquest projecte. Com que es tracta d'un projecte on hi ha una part de programari que consisteix en la implementació d'un algorisme de classificació d'imatges, les principals fonts de despeses són dues: llicències de programari i salaris. La resta del projecte consisteix en un estudi teòric, per tant entrarà dins els salaris.

4.1 Llicència

El programa utilitzat per desenvolupar l'algorisme de classificació d'imatges és MATLAB. Una llicència MATLAB individual té un cost de 800 euros/any, o 2000 euros indefinidament. Com que només es necessita una llicència per a aquest projecte, el cost total de les llicències és de 800 euros, ja que amb un any n'hi ha suficient.

4.2 Salaris

La segona font principal de despeses és el salari del desenvolupador d'algorismes. El desenvolupador d'aquest algorisme té un perfil d'enginyer de programari més jove. El salari típic per a un enginyer de programari menor és de 15 euros/h, i tenint en compte que el projecte es va iniciar el 26 de setembre de 2019 i va acabar el 26 de gener de 2020 i una dedicació setmanal de 20 hores, el cost de la càrrega de treball sencera del projecte és:

$$\text{Salari} = 15 \text{ setmanes} \times 20 \text{ hores/setmana} \times 15 \text{ EUR/hora} = 4500 \text{ EUR}$$

Per tant el cost total del projecte serà de:

$$\text{Cost Final} = 800 \text{ EUR} + 4500 \text{ EUR} = 5300 \text{ EUR}$$

Capítol 5

Conclusions i possibles millores

Aquest projecte tenia com a principal objectiu investigar tècniques de processament de senyal en grafs i mètodes per inferir la topologia dels grafs a partir de dades. La principal qüestió era si aquestes tècniques es podien aplicar a dades reals, i investigar si són d'utilitat en alguna aplicació. Després de trobar una base de dades de senyals EEG adequada, es va decidir investigar si els mètodes d'inferència de topologia de grafs podrien ser útils per a classificar aquests senyals EEG. Un cop feta l'experimentació hem comprovat que no classifica correctament.

Com s'observa en els resultats (3) tots els autovalors obtinguts de l'experimentació són molt semblants, per tant és difícil trobar un patró per classificar per grups.

Obtenir una base de dades per poder realitzar aquest experiment ha estat un dels punts més complicats, ja que és molt difícil trobar una base de dades, lliure, suficientment extensa i bona per poder fer l'experiment.

A continuació comentem els motius pels quals creiem que no hem obtingut els resultats esperats en l'experiment:

- ◇ El nombre de nodes obtinguts amb la base de dades és molt petit. El fet que només tinguem 5 nodes fa que sigui més complicat obtenir grafs diferents per diferents tipus de senyals.
- ◇ La classificació de les imatges pot ser que no fos la correcta. Hem classificat per temàtica de la imatge (animals marins, serps, gossos i pianos), però no tenim la certesa que fos la forma idònia de segmentar.

En base als arguments anteriors proposem les següents millores:

- ◇ Utilitzar una base de dades amb la qual obtinguem grafs de més nodes.
- ◇ Classificar les imatges en grups diferents, basant-nos en motius diferents per la classificació.
- ◇ Consultar amb un expert en senyals EEG per a veure si cal fer un preprocesat del senyal, per exemple, eliminar soroll o filtrar, abans d'utilitzar-los per a classificar.

Bibliografia

- [1] Mindbigdata "imagenet" of the brain. <http://www.mindbigdata.com/opendb/imagenet.html>, 2018.
- [2] Alba Pagès Zamora. Graph signal processing course. *Universitat Politècnica de Catalunya*, Chapter 1,2, June 2019.
- [3] Kaare Brandt Petersen, Michael Syskind Pedersen, et al. The matrix cookbook. *Technical University of Denmark*, 7(15):510, 2008.
- [4] Matriz de confusión matlab. <https://es.mathworks.com/help/stats/confusionmat.html>, 2019.

Capítol 6

Apèndixs

```
% Agafem l'arxiu que volem classificar.
db1 = xlsread('...');

% Normalitzem els valors amb la mitjana.
db = db1-(mean(db1'))';

% Calculem la matriu de covariància empírica.
S1 = (db * db') / length(db);
% Fem que la matriu sigui simètrica.
Sigma = ((S1 + S1')/2);

% Calculem la matriu d'adjacència utilitzant la fórmula de la
% correlació de Pearson.
for i = 1:5
    for j = 1:5
        R(i,j) = Sigma(i,j)/((Sigma(i,i)*Sigma(j,j))^(1/2));
    end
end

% Obtenim la Matriu d'adjacència amb el primer mètode.
A1 = R;
% Posem la diagonal de la matriu d'adjacència a 0.
A1 = A1- diag(diag(A1));
% Calculem el graf de la segona matriu d'adjacència i el representem.
AA1 = graph(A1);
```

```
% Calculem la Laplaciana del graf.
L1 = (diag(A1*ones(5,1)) - A1);

% Calculem els valors de la diagonal, si algun valor és 0 el deixem a 0.
% no li fem l'arrel quadrada.
D1n = diag(A1*ones(5,1));

for y = 1:5
    if D1n(y,y) == 0
        D1n(y,y) = 0;
    else
        D1n(y,y) = D1n(y,y)^(-1/2);
    end
end

% Calculem la Laplaciana normalitzada.
L1n = D1n * L1 * D1n;

% Calculem els autovalors de les Laplacianes.
lambda1 = sort(eig(L1))';
lambda1n = sort(eig(L1n))';

% Calculem la matriu inversa de la matriu de covariància empírica.
T1 = inv(Sigma);
% La Fem simètrica.
Teta= ((T1 + T1')/2);

% Obtenim la Matriu d'adjacència amb el segon mètode.
A2 = Teta;
% Posem la diagonal de la matriu d'adjacència a 0.
A2 = abs(A2- diag(diag(A2)));
% Calculem el graf de la segona matriu d'adjacència i el representem.
AA2 = graph(A2);

% Calculem la Laplaciana del graf.
L2 = diag(A2*ones(5,1)) - A2;

% Calculem els valors de la diagonal, si algun valor és 0 el deixem a 0.
% No li fem l'arrel quadrada.
D2n = diag(A2*ones(5,1));
```

```

for y = 1:5
    if D2n(y,y) == 0
        D2n(y,y) = 0;
    else
        D2n(y,y) = D2n(y,y)^(-1/2);
    end
end

% Calculem la Laplaciana normalitzada.
L2n = D2n * L2 * D2n;

% Calculem els autovalors de les Laplacianes .
lambda2 = sort(eig(L2))';
lambda2n = sort(eig(L2n))';

% Aquests vectors són les mitjanes de cada autovalor de les Laplacianes
% i Laplacianes normalitzades de cada un dels grups en els quals
% classifiquem.
% Han estat calculats anteriorment.
l2AM = [0 0.0029 0.0047 0.0073 0.0111];
l2S = [0 0.0028 0.0049 0.0074 0.0117];
l2G = [0 0.0026 0.0046 0.0071 0.0112];
l2P = [0 0.0030 0.00491 0.0076 0.0121];
l2nAM = [0 0.8412 1.1235 1.3812 1.6539];
l2nS = [0 0.8361 1.1134 1.3905 1.6598];
l2nG = [0 0.8336 1.1209 1.3854 1.6599];
l2nP = [0 0.8682 1.1066 1.3928 1.6323];

% Calculem l'error quadràtic mitjà dels autovalors de la Laplaciana
% respecte de la mitja d'autovalors de les Laplacianes de cada grup.
VAM = sum([(lambda2(1) - l2AM(1))^2 (lambda2(2) - l2AM(2))^2
    (lambda2(3) - l2AM(3))^2 (lambda2(4) - l2AM(4))^2
    (lambda2(5) - l2AM(5))^2]);
VS = sum([(lambda2(1) - l2S(1))^2 (lambda2(2) - l2S(2))^2
    (lambda2(3) - l2S(3))^2 (lambda2(4) - l2S(4))^2
    (lambda2(5) - l2S(5))^2]);
VG = sum([(lambda2(1) - l2G(1))^2 (lambda2(2) - l2G(2))^2
    (lambda2(3) - l2G(3))^2 (lambda2(4) - l2G(4))^2
    (lambda2(5) - l2G(5))^2]);

```

```

VP = sum([(lambda2(1) - l2P(1))^2 (lambda2(2) - l2P(2))^2
          (lambda2(3) - l2P(3))^2 (lambda2(4) - l2P(4))^2
          (lambda2(5) - l2P(5))^2]);

% Mirem quin és el valor mínim.
valors = [VAM VS VG VP];
miniml2 = min(valors);

% La posició sel valor minim serà el grup al qual pertany la senyal.
posl2 = find(valors == miniml2);

% Calculem l'error quadràtic mitjà dels autovalors de la Laplaciana
% normalitzada respecte de la mitja d'autovalors de les Laplacianes
% normalitzades de cada grup.
VAMn = sum([(lambda2n(1) - l2nAM(1))^2 (lambda2n(2) - l2nAM(2))^2
          (lambda2n(3) - l2nAM(3))^2 (lambda2n(4) - l2nAM(4))^2
          (lambda2n(5) - l2nAM(5))^2]);
VS n = sum([(lambda2n(1) - l2nS(1))^2 (lambda2n(2) - l2nS(2))^2
          (lambda2n(3) - l2nS(3))^2 (lambda2n(4) - l2nS(4))^2
          (lambda2n(5) - l2nS(5))^2]);
VG n = sum([(lambda2n(1) - l2nG(1))^2 (lambda2n(2) - l2nG(2))^2
          (lambda2n(3) - l2nG(3))^2 (lambda2n(4) - l2nG(4))^2
          (lambda2n(5) - l2nG(5))^2]);
VP n = sum([(lambda2n(1) - l2nP(1))^2 (lambda2n(2) - l2nP(2))^2
          (lambda2n(3) - l2nP(3))^2 (lambda2n(4) - l2nP(4))^2
          (lambda2n(5) - l2nP(5))^2]);

% Mirem quin es el valor mínim.
valorsn = [VAMn VS n VG n VP n];
miniml2n = min(valorsn);

% La posició del valor mínim serà el grup al qual pertany el senyal.
posl2n = find(valorsn == miniml2n);

```



Universitat Politècnica de Catalunya

Barcelona, 2020